Amendments to the Claims

Please amend the claims according to the following directions. Please replace all prior versions and listings of claims in this application with the following list of claims:

1. (currently amended) A method for synchronizing parallel texture pipelines ~~in a graphics engine~~, comprising:

loading an array of ~~polygon~~ state variables for a polygon into an accumulation portion of a plurality of ~~sets of~~ parallel texture pipelines ~~pipeline state variable queues~~; and then

simultaneously enabling a ~~texture~~ processing portion of a number of the ~~sets of state variable queues~~ parallel texture pipelines, said number corresponding to a number of parallel texture operations indicated by the ~~polygon~~ loaded array of state variables.

2. (currently amended) The method of claim 1, wherein the loading further comprises for each parallel texture pipeline ~~state variable queue~~:

receiving the ~~polygon~~ array of state variables in ~~a state variable~~ an accumulator; and

~~copying~~ transferring the received ~~polygon~~ array of state variables to a ~~state variable~~ latching register.

3. (currently amended) The method of claim 2, wherein the ~~copying~~ transferring is performed substantially simultaneously for each parallel texture pipeline, prior to ~~processing each polygon~~ the enabling.

4. (currently amended) The method of claim 1, further comprising disabling the ~~texture~~ processing portions of the remaining ~~sets of state variable queues~~ non-enabled parallel texture pipelines.

5. (currently amended) The method of claim 4, further comprising removing power to the ~~pipelines corresponding to the~~ disabled ~~texture~~ processing portions.

6. (currently amended) A method of synchronizing ~~multiple~~ parallel texture pipelines, comprising:

accumulating ~~state variable data~~ in an accumulation portion of each parallel texture pipeline a set of state variable data for a polygon;

determining a number N of textures to be applied to the polygon, based on the set of state variable data;

for ~~a predetermined number~~ each N ~~of the~~ parallel texture ~~pipelines~~ pipeline, simultaneously advancing the accumulated set of state variable data from the accumulation portion to a downstream processing portion ~~succeeding portions of the texture pipeline, N representing a number of textures to be applied to polygon data~~.

7. (currently amended) The method of claim 6, further comprising disabling downstream processing portions of the remaining parallel texture pipelines.

8. (original) The method of claim 6, wherein the accumulating comprises:

receiving new state variable data, the new state variable data being defined differentially with respect to old state variable data previously accumulated, and

evicting obsolete elements of the old state variable data in favor of the new state variable data.

9. (currently amended) A control method for a texture processing system having multiple parallel texture pipelines, comprising:

in each parallel texture pipeline ~~state variable queue~~, accumulating ~~an array of~~ state variable data for a polygon in ~~a~~ an accumulation register, elements of the array of ~~the~~ state variable data being received over a plurality of clock cycles ~~as a plurality of data units~~;

if the texture processing system switches texture modes, transitioning from a first number of active texture pipelines to a second number of active texture pipelines; and then substantially simultaneously, in each of ~~a number of texture pipelines corresponding to~~ the second number of active texture pipelines,

advancing the array of state variable data from the ~~respective registers~~ accumulation register to a remainder ~~of the respective state variable queues~~ processing portion~~, and~~

~~disabling the remaining texture pipelines and portions of associated state variable queues~~.

10. (currently amended) The method of claim 9, further comprising:

enabling power to the second number of active texture pipelines; and

disabling power to the remaining texture pipelines.

11. (currently amended) A texture processing system, comprising:

a plurality of parallel texture pipelines, each parallel texture pipeline having a state variable queue, each state variable queue comprising an accumulation register for accumulating a set of state variables for a polygon, a latching register coupled to the accumulation register, and a state variable FIFO buffer coupled to the latching register;

~~a plurality of state variable queues, at least one state variable queue provided for each texture pipeline, the state variable queues each comprising an accumulation register and a latching register coupled to a series of state variable queue processing stages;~~ and

a controller having a data output coupled to the plurality of accumulation registers, the controller having a first control output ~~adapted to disable at least one of the series of state variable queue processing stages~~ to trigger the advance of the set of state variables from the plurality of accumulation registers to the plurality of latching registers, and the controller having a second control output to trigger the advance of the set of state variables from a number of the latching registers to their corresponding state variable FIFO buffers, said number representing a number of parallel textures indicated by the set of state variables.

12. (currently amended) The texture processing system of claim 11, wherein said controller ~~has~~ further comprises a data input ~~adapted~~ to receive a state variable from a programming source.

13. (cancelled).

14. (cancelled).

15. (cancelled).

16. (cancelled).

17. (cancelled).

18. (new) A texture processor, comprising:

a first parallel texture pipeline having a first state variable queue including a first accumulator for receiving polygon state variables, a first latching register interoperably connected to the first accumulator, and a first texture processing FIFO interoperably connected to the first latching register;

a second parallel texture pipeline having a second state variable queue including a second accumulator for receiving the polygon state variables, a second latching register interoperably connected to the second accumulator, and a second texture processing FIFO interoperably connected to the second latching register;

a controller having an input for receiving a stream of data for a polygon including the polygon state variables, said controller having a first output for forwarding the polygon state variables to the first accumulator and the second accumulator, said controller having a second output for a state variable advance signal to trigger substantially simultaneously the transfer of the polygon state variables from the first accumulator to the first latching register and the transfer of the polygon state variables from the second accumulator to the second latching register, said controller having a third output for a first pipeline advance signal to transfer the polygon state variables from the first latching register to the first texture processing FIFO, and said controller having a fourth output for a second pipeline advance signal to transfer the polygon state variables from the second latching register to the second texture processing FIFO only when the polygon state variables indicate multiple parallel texture operations.

19. (new) A computer system, comprising:

a processor coupled to a bus;

a system memory in communication with the bus; and

a graphics unit comprising the texture processor of claim 18.

20. (new) A method for synchronizing parallel texture pipelines, comprising:

accumulating a set of state variables for a polygon in a first state variable accumulator of a first texture pipeline and in a second state variable accumulator of a second texture pipeline;

simultaneously transferring the set of state variables from the first state variable accumulator to a first state variable latching register of the first texture pipeline and from the

second state variable accumulator to a second state variable latching register of the second texture pipeline;

advancing the set of state variables from the first state variable latching register to a remaining portion of the first texture pipeline; and

advancing the set of state variables from the second state variable latching register to a remaining portion of the second texture pipeline only when the set of state variables indicate two parallel texture operations for the polygon.